10/673,288

1-30-4

01807.002334. PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	) :	Examiner: Not Yet Known
FRÉDÉRIC LEHOBEY ET AL.	) :	Group Art Unit: 2133
Application No.: 10/673,288	) :	
Filed: September 30, 2003	) :	
For: METHODS AND DEVICES FOR DECODING ONE-POINT ALGEBRAIC GEOMETRIC CODES	) :	January 29, 2004

MAIL STOP - MISSING PARTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

SUBMISSION OF PRIORITY DOCUMENT

Sir:

In support of Applicants' claim for priority under 35 U.S.C. § 119, enclosed is a certified copy of the following French application:

0212069 filed September 30, 2002.

Applicants' undersigned attorney may be reached in our New York office by telephone at (212) 218-2100. All correspondence should continue to be directed to our address given below.

Respectfully submitted,

Raymond D. Perna

Attorney for Applicants

Registration No. 44,063

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200

R E P U B L I Q U E F R A N Ç A I S E

10/673,288
Sau 2128

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le 30 SEP. 2003

Pour le Directeur général de l'Institut
national de la propriété industrielle
Le Chef du Département des brevets

M Planche

Martine PLANCHE

INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
SIEGE
26 bis, rue de Saint Petersbourg
75800 PARIS cedex 08
Téléphone : 33 (0)1 53 04 53 04
Télécopie : 33 (0)1 53 04 45 23
www.inpi.fr

ETABLISSEMENT PUBLIC NATIONAL CREE PAR LA LOI N° 51-444 DU 19 AVRIL 1951

1er dépôt

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11354*02

REQUÊTE EN DÉLIVRANCE
page 1/2

BR1

Cet imprimé est à remplir lisiblement à l'encre noire — DB 540 W / 010301

Réservé à l'INPI

REMISE DES PIÈCES
DATE
LIEU 30 SEPT 2002
75 INPI PARIS
N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0212069
DATE DE DÉPÔT ATTRIBUÉE PAR L'INPI 30 SEP. 2002

Vos références pour ce dossier *(facultatif)* BIF023147/DM/LJH

1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE

RINUY, SANTARELLI
14, avenue de la Grande Armée
75017 PARIS

Confirmation d'un dépôt par télécopie ☐ N° attribué par l'INPI à la télécopie

2 NATURE DE LA DEMANDE — Cochez l'une des 4 cases suivantes

Demande de brevet	☒
Demande de certificat d'utilité	☐
Demande divisionnaire	☐
Demande de brevet initiale	N° Date
ou demande de certificat d'utilité initiale	N° Date
Transformation d'une demande de brevet européen *Demande de brevet initiale*	☐ N° Date

3 TITRE DE L'INVENTION (200 caractères ou espaces maximum)

Procédés et dispositifs pour le décodage des codes de géométrie algébrique à un point

4 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE

Pays ou organisation
Date N°
Pays ou organisation
Date N°
Pays ou organisation
Date N°

☐ **S'il y a d'autres priorités, cochez la case et utilisez l'imprimé «Suite»**

5 DEMANDEUR (Cochez l'une des 2 cases) ☒ **Personne morale** ☐ **Personne physique**

Nom ou dénomination sociale		CANON KABUSHIKI KAISHA
Prénoms		
Forme juridique		Société de droit Japonais
N° SIREN		
Code APE-NAF		
Domicile ou siège	Rue	30-2, Shimomaruko 3-chome, Ohta-ku,
	Code postal et ville	Tokyo
	Pays	JAPON
Nationalité		JAPONAISE
N° de téléphone *(facultatif)*		N° de télécopie *(facultatif)*
Adresse électronique *(facultatif)*		

☐ **S'il y a plus d'un demand ur, cochez la case et utilisez l'imprimé «Suite»**

Remplir impérativement la 2ème page

1er dépôt

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

REQUÊTE EN DÉLIVRANCE
page 2/2

R2

Réservé à l'INPI

REMISE DES PIÈCES
DATE
LIEU 30 SEPT 2002
75 INPI PARIS
N° D'ENREGISTREMENT
NATIONAL ATTRIBUÉ PAR L'INPI 0212069

DB 540 W /300301

Vos références pour ce dossier : *(facultatif)*	BIF023147/DM/LJH
6 MANDATAIRE	
Nom	
Prénom	
Cabinet ou Société	RINUY, SANTARELLI
N °de pouvoir permanent et/ou de lien contractuel	
Adresse' Rue	14 AVENUE DE LA GRANDE ARMEE
Code postal et ville	7 5 0 1 7 PARIS
N° de téléphone *(facultatif)*	01 40 55 43 43
N° de télécopie *(facultatif)*	
Adresse électronique *(facultatif)*	
7 INVENTEUR (S)	
Les inventeurs sont les demandeurs	☐ Oui ☒ Non **Dans ce cas fournir une désignation d'inventeur(s) séparée**
8 RAPPORT DE RECHERCHE	**Uniquement pour une demande de brevet (y compris division et transformation)**
Établissement immédiat ou établissement différé	☒ ☐
Paiement échelonné de la redevance	**Paiement en deux versements, uniquement pour les personnes physiques** ☐ Oui ☐ Non
9 RÉDUCTION DU TAUX DES REDEVANCES	**Uniquement pour les personnes physiques** ☐ Requise pour la première fois pour cette invention *(joindre un avis de non-imposition)* ☐ Requise antérieurement à ce dépôt *(joindre une copie de la décision d'admission pour cette invention ou indiquer sa référence)* :
Si vous avez utilisé l'imprimé «Suite», indiquez le nombre de pages jointes	

10 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE (Nom et qualité du signataire)

Bruno QUANTIN N°92.1206
RINUY, SANTARELLI

VISA DE LA PRÉFECTURE OU DE L'INPI

L. GUICHET

La loi n°78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés s'applique aux réponses faites à ce formulaire. Elle garantit un droit d'accès et de rectification pour les données vous concernant auprès de l'INPI.

La présente invention concerne les systèmes de communication dans lesquels, afin d'améliorer la fidélité de la transmission, les données à transmettre sont soumises à un codage de canal. Elle concerne plus particulièrement un procédé de décodage, ainsi que les dispositifs et appareils destinés à mettre en œuvre ce procédé.

On rappelle que le codage dit « de canal » consiste, quand on forme les « mots de code » envoyés au récepteur, à introduire une certaine redondance dans les données à transmettre. Plus précisément, on transmet, au moyen de chaque mot de code, un nombre prédéterminé k de symboles d'information choisis au sein d'un « alphabet » prédéterminé de taille finie q ; on ajoute à ces k symboles d'information un nombre $(n-k)$ de symboles dits de « parité », prélevés dans le même alphabet, de manière à former des mots de code $\underline{c} = (c_1, c_2, \ldots, c_n)$ de longueur n ; l'ensemble des règles de calcul des symboles de parité en fonction des symboles d'information définit un « code », ou « procédé de codage », de « dimension » k et de « longueur » n, ainsi caractérisé par un certain ensemble de mots de code constituant une sorte de dictionnaire. On peut, de façon commode, définir un code au moyen d'une matrice H, de dimension $(n-k)$ x n, dite « matrice de parité » : un mot $\underline{c}$ de longueur n donné est un mot de code si, et seulement s'il vérifie la relation : $H \cdot \underline{c}^T = 0$ (où l'exposant T indique la transposition).

Au niveau du récepteur, le procédé de décodage associé exploite alors judicieusement cette redondance pour détecter d'éventuelles erreurs de transmission et si possible les corriger. Il y a erreur de transmission si la différence $\underline{e}$ entre un mot reçu $\underline{r}$, et le mot de code $\underline{c}$ correspondant envoyé par l'émetteur, est non-nulle.

Plus précisément, le décodage se fait en deux étapes principales.

La première étape consiste à associer au mot reçu un « mot de code associé ». Pour ce faire, le décodeur calcule d'abord le « syndrome d'erreur » $\underline{s} = H \cdot \underline{r}^T = H \cdot \underline{e}^T$. Si le syndrome est nul, on supposera qu'il n'y a pas eu d'erreur de transmission, et le « mot de code associé » sera alors simplement pris égal au mot reçu. Si ce n'est pas le cas, on en déduit que certains symboles dans le mot reçu sont erronés, et l'on met alors en œuvre un

algorithme de correction destiné à estimer la valeur de l'erreur $\underline{e}$; l'algorithme va ainsi fournir une valeur estimée $\underline{\hat{e}}$ de manière à ce que $(\underline{r} - \underline{\hat{e}})$ soit un mot de code, qui constituera alors le « mot de code associé ».

La seconde étape consiste simplement à inverser le procédé de codage, c'est-à-dire à retirer les symboles de redondance du « mot de code associé » pour retrouver les symboles d'information initiaux.

L'invention concerne plus particulièrement la première de ces deux étapes, et les conditions de mise en œuvre de l'algorithme de correction.

Un algorithme de correction a pour tâche d'associer au mot reçu le mot de code situé à la distance de Hamming la plus courte de ce mot reçu, la « distance de Hamming » étant, par définition, le nombre d'emplacements où deux mots de même longueur possèdent un symbole différent. Chaque code offre donc une capacité de correction d'erreurs qui est limitée par la distance de Hamming la plus petite entre deux mots quelconques de ce code, que l'on appelle la « distance minimale » du code d ; plus précisément, quand l'algorithme de correction choisi est chargé de trouver la position des erreurs éventuelles dans un mot reçu quelconque, et de fournir un symbole de remplacement pour chacune de ces positions, on est sûr de pouvoir corriger au mieux INT[(d-1)/2] erreurs pour un code de distance minimale d (« INT » désigne la partie entière). Si le mot reçu contient un nombre d'erreurs strictement supérieur à INT[(d-1)/2], l'algorithme sera dans certains cas capable de proposer une correction, mais il sera alors évidemment très douteux que cette correction soit la bonne, c'est-à-dire que le mot de code associé soit bien le mot de code envoyé par l'émetteur.

La capacité d'un algorithme de correction à pouvoir proposer une correction d'un mot reçu est fidèlement représentée par la formule :

$$2t \leq \Delta,$$

où t est le nombre de symboles erronés dans le mot reçu, et Δ est un entier strictement positif que nous appellerons le « pouvoir de résolution » de l'algorithme. Si la valeur de ($2t$) est inférieure ou égale au pouvoir de résolution, l'algorithme de correction sera capable de corriger le mot reçu. Si la valeur de ($2t$) est supérieure au pouvoir de résolution, l'algorithme pourra :

- soit échouer purement et simplement dans sa tentative de correction,

- soit être capable de proposer une correction du mot reçu ; dans ce cas, si l'on accepte cette correction, on s'expose au risque qu'elle soit erronée, c'est-à-dire que le mot de code proposé ne soit pas, en fait, le mot envoyé, ce risque étant évidemment d'autant plus prononcé que $(2t)$ est grand par rapport à Δ.

Compte tenu des considérations ci-dessus concernant la distance minimale d du code, on dira que l'algorithme considéré est « maximal » si

$$\Delta = d - 1,$$

et « sub-maximal » si

$$\Delta < d - 1.$$

Parmi les procédés de codage connus, on peut citer les « codes de Reed-Solomon », qui sont réputés pour leur efficacité. Ces codes, toutefois, présentent la particularité que la longueur n des mots de code est nécessairement inférieure ou égale à la taille q de l'alphabet des symboles. De ce fait, si l'on souhaite disposer d'un code de Reed-Solomon ayant des mots de code de grande longueur, on doit envisager de larges valeurs de q, ce qui conduit à des mises en œuvre coûteuses au niveau des calculs et de la mémorisation. De plus, de larges valeurs de q sont parfois inadaptées à l'application technique envisagée.

Or dans les supports d'information modernes, par exemple dans les enregistrements de CD (« *compacts discs* ») et de DVD (« *digital video discs* »), on cherche à accroître la densité d'information. Quand un tel support est affecté par un défaut physique tel qu'une éraflure, un nombre important de symboles d'information peuvent être rendus illisibles. On peut toutefois remédier à ce problème en utilisant des mots de code de très grande longueur. C'est pourquoi l'on a cherché à construire des codes offrant de manière naturelle une plus grande longueur de mots que les codes de Reed-Solomon.

On a notamment proposé récemment des codes dits « codes de géométrie algébrique » ou « codes de Goppa géométriques » (voir par exemple « *Algebraic Geometric Codes* », par J.H. van Lint, dans « *Coding Theory and Design Theory* », 1ère partie, *IMA Volumes Math. Appl.*, volume 21, Springer-

Verlag, Berlin, 1990). Ces codes sont construits à partir de courbes algébriques définies sur un alphabet à q éléments structuré en corps de Galois. Un paramètre important d'une telle courbe est son « genre » g. Dans le cas particulier où la courbe est une simple droite (le genre g est alors nul), le code de géométrie algébrique se réduit à un code de Reed-Solomon. Dans certains cas, les codes de géométrie algébrique permettent d'atteindre une longueur égale à $\left(q+2g\sqrt{q}\right)$, qui peut être très élevée ; par exemple, avec une taille d'alphabet égale à 256 et un genre égal à 120, on obtient des mots de code de longueur 4096.

Les codes de géométrie algébrique sont, comme on l'a dit, avantageux quant à la longueur des mots de code, mais ils présentent l'inconvénient de requérir (en tous cas dans l'état actuel des connaissances) des algorithmes de décodage assez complexes, et donc assez coûteux en termes d'équipements (logiciel et/ou matériel) et de temps de traitement. Cette complexité est en fait plus ou moins grande selon l'algorithme considéré, une plus grande complexité étant en principe le prix à payer pour accroître la capacité de correction d'erreurs du décodeur (voir par exemple l'article de Tom Høholdt et Ruud Pellikaan intitulé « *On the Decoding of Algebraic-Geometric Codes* », *IEEE Trans. Inform. Theory*, vol. 41 n° 6, pages 1589 à 1614, novembre 1995).

Il est à noter que pour ces algorithmes, on ne dispose que d'une borne inférieure de leur pouvoir de résolution Δ, sauf dans le cas « trivial » de l'algorithme maximal de correction des codes de Reed-Solomon (appelé « algorithme de Berlekamp-Massey »), pour lequel le pouvoir de résolution est précisément connu et vaut $\Delta = n-k$. Une généralisation de cet algorithme aux codes de géométrie algébrique de genre non-nul, appelée algorithme « de base » a été proposée par A.N. Skorobogatov et S.G. Vlăduț dans l'article intitulé « *On the Decoding of Algebraic-Geometric codes* », *IEEE Trans. Inform. Theory*, vol. 36 n° 5, pages 1051 à 1060, novembre 1990) ; cet algorithme offre un pouvoir de résolution au moins égal à $\Delta = n-k-2g$.

Or la distance minimale d pour un code de géométrie algébrique est au moins égale à $(n-k+1-g)$. Il est donc clair que l'algorithme de base est

« sub-maximal », et cela d'autant plus que le genre g de la courbe algébrique est grand. Dans le but d'améliorer le pouvoir de résolution, Skorobogatov et Vlăduț ont proposé, dans le même article cité ci-dessus, une version « modifiée » de l'algorithme « de base ». Cet algorithme « modifié » présente un pouvoir de résolution au moins égal à $\Delta = n - k - g - s$, où s est un paramètre dépendant de la courbe algébrique choisie, et pouvant d'ailleurs être quelquefois nul (c'est par exemple le cas pour les courbes algébriques dites « hyperelliptiques »).

L'algorithme de base procède essentiellement en trois étapes :

1) on construit une « matrice des syndromes » S, de dimension $(n-k)$ x $(n-k)$, dont chaque coefficient S_{ij}, où j est inférieur ou égal à une valeur « frontière » $w(i)$, est égal à une combinaison linéaire judicieusement choisie des éléments s_ν ($\nu = 1,2,\ldots,n-k$) du syndrome $\underline{s}$, les coefficients S_{ij} au-delà de la frontière restant indéterminés,

2) on identifie la *position* des erreurs dans le mot reçu, en résolvant un certain nombre d'équations linéaires dont les coefficients sont pris dans la matrice des syndromes S, et

3) on *corrige* les symboles erronés du mot reçu dont on connaît dès lors la position.

La modification introduite par l'algorithme de base modifié consiste en un nouveau mode opératoire pour la deuxième étape de l'algorithme. Plus précisément, on considère, pour tout entier μ compris entre 1 et $(n-k)$, le système d'équations linéaires

$$\sum_{i=1}^{\mu} l_i S_{ij} = 0, \quad \text{pour } j = 1,2,\ldots,w(\mu), \tag{1}$$

où les valeurs des inconnues l_i sont à trouver dans le même alphabet de symboles que les éléments des mots de code. On recherche alors un entier λ_0, qui est la plus petite valeur de μ pour laquelle un tel système possède une solution non-triviale, c'est-à-dire une solution où les coefficients l_i ne sont pas tous nuls.

Skorobogatov et Vlăduț enseignent donc de considérer successivement, et de manière indépendante, des sous-matrices de S de dimension $\mu \times w(\mu)$, d'abord pour μ égal à 1, puis pour μ égal à 2, et ainsi de suite, jusqu'à en trouver une dont les lignes sont linéairement dépendantes.

Or une question importante sur le plan pratique et que l'on doit se poser à propos de n'importe quel algorithme de calcul, est celle de sa complexité, c'est-à-dire du nombre d'opérations arithmétiques qu'il requiert. On peut montrer que la résolution de l'algorithme de base modifié, tel que décrit succinctement ci-dessus, requiert de l'ordre de n^4 opérations arithmétiques (dans l'alphabet des symboles), où n est, on le rappelle, la longueur des mots de code. Or la complexité de l'algorithme de base n'est que de l'ordre de n^3. Ainsi donc, l'augmentation du pouvoir de résolution selon cette approche s'est faite au prix d'une augmentation de la complexité.

La présente invention a notamment pour but de trouver un algorithme de décodage qui ait un pouvoir de résolution au moins égal à celui de l'algorithme de base modifié pour le même code, mais dont la complexité soit aussi faible que possible, et croissant au plus comme n^3.

L'article de I.M. Duursma intitulé « *Algebraic Decoding using Special Divisors* » (IEEE Transactions on Information Theory, vol. 39, n° 2, pages 694 à 698, 1993) propose un perfectionnement (appelé « algorithme modifié étendu » par l'auteur) de l'algorithme de base modifié, destiné à en réduire la complexité. Duursma montre qu'il peut exister en général plusieurs valeurs de μ pour lesquelles le système (1) possède une solution non-triviale, et pour lesquelles l'algorithme de décodage complet a le même pouvoir de résolution que l'algorithme de base modifié. Nous appellerons « dimension résolvante » λ toute valeur de μ ayant cette propriété, de sorte que la valeur λ_0 mentionnée ci-dessus est redéfinie comme étant la plus petite de ces dimensions résolvantes. De plus, il ressort de l'article cité qu'il existe au moins une dimension résolvante dont la valeur est supérieure ou égale à un certain entier, que nous désignerons par le « minimum de Duursma » μ_D (pour la méthode de calcul de μ_D, on pourra se référer à cet article). Nous désignerons par « dimension étendue » λ_D la plus

petite de ces dimensions résolvantes supérieures ou égales à μ_D (λ_D pouvant d'ailleurs le cas échéant être égale à la plus petite dimension résolvante λ_0).

Par conséquent, selon l'algorithme modifié étendu, on cherche à résoudre le système (1) en considérant, comme dans l'algorithme de base modifié, des valeurs successives de μ, mais en commençant cette recherche à $\mu = \mu_D$. La complexité de l'algorithme modifié étendu croît comme gn^3 ; or, de manière générale, les genres g des courbes algébriques utilisées pour le codage peuvent être de grands nombres : en effet, pour pouvoir construire des codes dont la longueur est de plus en plus grande, on est souvent amené à utiliser des courbes algébriques dont les genres sont de plus en plus grands.

Des algorithmes de décodage des codes de géométrie algébrique dont la complexité croît comme n^3 ont été proposés en utilisant une approche différente de celle de l'algorithme modifié.

L'article de G.-L. Feng et T.R.N. Rao intitulé « *Decoding Algebraic-Geometric Codes up to the Designed Minimum Distance* » (IEEE Transactions on Information Theory, vol. 39, n° 1, janvier 1993) divulgue un tel algorithme. Dans celui-ci, on résout le système d'équations linéaires de l'algorithme de base (voir étape 2) ci-dessus), après avoir déterminé selon une certaine règle les éléments *a priori* inconnus de la matrice des syndromes (voir étape 1) ci-dessus). Mais le calcul de ces éléments complémentaires de la matrice S est compliqué, et par conséquent, même si la complexité de cet algorithme croît comme n^3, le nombre d'opérations effectif (égal à $C \cdot n^3$, où C est une constante de valeur élevée) est très grand.

L'article de R. Kötter intitulé « *Fast Generalized Minimum-Distance Decoding of Algebraic-Geometry and Reed-Solomon Codes* » (IEEE Transactions on Information Theory, vol. 42, n° 3, mai 1996) divulgue lui aussi un tel algorithme. Dans celui-ci, on recherche la solution de *tous* les systèmes d'équations linéaires conformes à l'algorithme de base, alors *qu'un seul* suffirait. Il en résulte ici aussi une complexité égale à $C' \cdot n^3$, où C' est une constante de valeur élevée.

Or les auteurs de la présente invention ont découvert que, contrairement à ce que l'on pouvait penser de prime abord, il est en fait

possible -- du moins en ce qui concerne les codes de géométrie algébrique dits « à un point » -- de définir un algorithme de décodage ayant une complexité proportionnelle à n^3, avec une constante de proportionnalité de l'ordre de 1, en conservant la philosophie générale, et le pouvoir de résolution, de l'algorithme de base modifié de Skorobogatov et Vlăduț (pour une définition des codes de géométrie algébrique « à un point », on pourra consulter par exemple l'article de Tom Høholdt et Ruud Pellikaan cité ci-dessus). L'invention le démontre en proposant un algorithme où les calculs sont organisés de manière judicieuse dans ce but. Plus précisément, l'invention enseigne comment il est possible, quand on cherche à résoudre le système d'équations (1) pour une valeur μ_0 donnée de μ, de prendre en compte certaines informations résultant des tentatives infructueuses de résolution du système (1) pour les valeurs de μ inférieures à μ_0 (sauf pour $\mu_0 = 1$) ; en exploitant ces informations, l'invention permet de réduire considérablement la complexité des calculs requis lors de la tentative de résolution pour la valeur μ_0.

L'invention concerne donc, selon un premier aspect, un procédé de décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, dans lequel, afin d'identifier la position des erreurs dans un mot reçu, on définit la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu, ledit procédé étant remarquable en ce qu'il comporte des étapes numérotées u de construction de matrices, au cours desquelles l'on construit des matrices S^u en commençant par $S^1 = S$, en ce que chaque matrice S^u pour $u > 1$ est obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis
- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce que la construction de matrices s'arrête lorsque :

- soit $S^u{}_{uj} = 0$ pour tout j compris entre 1 et $w(u)$,
- soit il existe un entier $u^* \leq (u-1)$ tel que $S^{u^*}{}_{u^*j} = 0$ pour tout j compris entre 1 et $w(u)$.

Par « manipulation linéaire » sur les lignes, on entend les remplacements d'une ligne par une combinaison linéaire de celle-ci avec une ou plusieurs autres lignes.

Le principe général du procédé selon l'invention s'inspire de l'algorithme dit du « pivot de Gauss », en prenant bien soin de n'opérer que sur les éléments connus de la matrice S, c'est-à-dire ceux situés en deçà de la « frontière » représentée par la fonction w. Comme décrit en détail ci-dessous, cette approche fournit les coefficients l_i associés à une dimension résolvante λ avec une complexité du calcul qui n'est que de l'ordre de n^3.

Naturellement, le procédé selon la présente invention permet également de déterminer des valeurs de dimensions résolvantes. L'invention concerne donc aussi, selon le même premier aspect, un procédé de décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, dans lequel, afin d'identifier la position des erreurs dans un mot reçu, on définit la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu, ledit procédé étant remarquable en ce qu'il comporte des étapes numérotées u de construction de matrices, au cours desquelles l'on construit des matrices S^u, en commençant par $S^1 = S$, en ce que chaque matrice S^u pour $u > 1$ est obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis
- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce que la dernière étape est :

- soit l'étape de numéro $u = \lambda$, si l'on détermine un entier λ tel que $S^{\lambda}{}_{\lambda j} = 0$ pour tout j compris entre 1 et $w(\lambda)$,
- soit l'étape de numéro $u = (\lambda - 1)$, si l'on détermine un entier λ et un entier u^*, avec $u^* < \lambda$, tels que $S^{u^*}{}_{u^* j} = 0$ pour tout j compris entre 1 et $w(\lambda)$.

Comme on le voit, la prise en compte de la « frontière » w joue un rôle central dans l'invention. Une propriété notable des codes de géométrie algébrique est que la fonction w est toujours décroissante (au sens large) quand la

matrice de parité H est construite de façon canonique (si ce n'est pas le cas, il suffira de réarranger en conséquence l'ordre des lignes de la matrice S avant de mettre en œuvre le procédé de décodage selon l'invention). Soit donc u_{max} l'indice de la première ligne pour laquelle $w(u_{max})$ est inférieur à u_{max}. Le système (1) pour $\mu = u_{max}$ comporte alors plus d'inconnues que d'équations : il possède donc évidemment une solution non-triviale. On en déduit qu'il existe nécessairement une dimension résolvante de valeur inférieure ou égale à u_{max}. Il est par conséquent inutile dans les calculs de conserver, dans les matrices S^u, les lignes d'indice supérieur à u_{max}. On réduira donc de préférence la complexité des calculs et le stockage en tronquant à u_{max} le nombre de lignes de chaque matrice S^u.

Selon l'algorithme de base modifié, ou selon l'algorithme modifié étendu, on forme, après avoir calculé un jeu de coefficients l_i, le « polynôme de localisation d'erreurs », dont les racines servent à trouver la position des erreurs dans le mot reçu. La présente invention est, avantageusement, compatible avec cette méthode connue pour localiser les erreurs de transmission.

Selon des caractéristiques particulières, le nombre de colonnes de chaque matrice S^u est tronqué à $w(u)$. L'algorithme selon l'invention fournit alors la plus petite dimension résolvante λ_0 et les coefficients l_i associés. Un avantage de ce mode de réalisation est que le polynôme de localisation d'erreurs qui en résulte possède un nombre minimal de coefficients.

Selon d'autres caractéristiques particulières, le nombre de colonnes de chaque matrice S^u est tronqué à $w(\mu_D)$ pour u compris entre 1 et le minimum de Duursma μ_D, et à $w(u)$ pour (le cas échéant) u supérieur à μ_D. L'algorithme selon l'invention fournit alors la dimension étendue λ_D et les coefficients l_i associés. Un avantage de ce mode de réalisation est d'entraîner une réduction du stockage, puisque les matrices S^u peuvent ne comporter que $w(\mu_D)$ colonnes pour u compris entre 1 et μ_D, et $w(u)$ colonnes pour u supérieur à μ_D.

Selon un autre de ses aspects, l'invention concerne divers dispositifs.

Elle concerne ainsi, premièrement, un dispositif de correction d'erreurs pour le décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, destiné à identifier la position des erreurs dans un mot reçu, et comprenant des moyens pour définir la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu, ledit dispositif de correction d'erreurs (107) étant remarquable en ce qu'il comprend en outre des moyens pour construire des matrices S^u numérotées u, avec $S^1 = S$, chaque matrice S^u pour $u > 1$ étant obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis
- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce qu'il comporte des moyens pour arrêter la construction de matrices lorsque :

- soit $S^u{}_{uj} = 0$ pour tout j compris entre 1 et $w(u)$,
- soit il existe un entier $u^* \leq (u-1)$ tel que $S^{u^*}{}_{u^*j} = 0$ pour tout j compris entre 1 et $w(u)$.

De préférence, ce dispositif de correction d'erreurs comprendra en outre des moyens pour tronquer à u_{max} le nombre de lignes de chaque matrice S^u, où u_{max} est le plus petit entier i pour lequel $w(i)$ est inférieur à i.

Selon des caractéristiques particulières, le dispositif de correction d'erreurs comprend en outre des moyens pour tronquer à $w(u)$ le nombre de colonnes de chaque matrice S^u.

Selon d'autres caractéristiques particulières, le dispositif de correction d'erreurs comprend en outre des moyens pour tronquer à $w(\mu_D)$ pour u compris entre 1 et le minimum de Duursma μ_D, et à $w(u)$ pour (le cas échéant) u supérieur à μ_D, le nombre de colonnes de chaque matrice S^u.

Les avantages de ces dispositifs de correction d'erreurs sont essentiellement les mêmes que ceux des procédés corrélatifs décrits succinctement ci-dessus.

L'invention concerne aussi, deuxièmement, un décodeur comprenant :

- au moins un dispositif de correction d'erreurs tel que décrit succinctement ci-dessus, et
- au moins une unité de suppression de la redondance.

L'invention vise également :

- un appareil de réception de signaux numériques codés comprenant un décodeur tel que décrit succinctement ci-dessus, ainsi que des moyens pour démoduler lesdits signaux numériques codés,
- un système informatique comprenant un décodeur tel que décrit succinctement ci-dessus, et comprenant en outre au moins un disque dur, et au moins un moyen de lecture de ce disque dur,
- un moyen de stockage de données inamovible comportant des instructions de code de programme informatique pour l'exécution des étapes de l'un quelconque des procédés succinctement exposés ci-dessus,
- un moyen de stockage de données partiellement ou totalement amovible, comportant des instructions de code de programme informatique pour l'exécution des étapes de l'un quelconque des procédés succinctement exposés ci-dessus, et
- un programme d'ordinateur, contenant des instructions telles que, lorsque ledit programme commande un dispositif de traitement de données programmable, lesdites instructions font que ledit dispositif de traitement de données met en œuvre l'un des procédés succinctement exposés ci-dessus.

Les avantages offerts par ce décodeur, cet appareil de réception, ce système informatique, ces moyens de stockage de données et ce programme d'ordinateur sont essentiellement les mêmes que ceux offerts par les procédés selon l'invention.

D'autres aspects et avantages de l'invention apparaîtront à la lecture de la description détaillée ci-dessous de modes de réalisation particuliers, donnés à titre d'exemples non limitatifs. La description se réfère aux dessins qui l'accompagnent, dans lesquels :

- la figure 1 est un schéma synoptique d'un système de transmission d'informations utilisant un codage de canal selon l'invention,

- la figure 2 est une liste de monômes qui forment une base pour un espace vectoriel de fonctions à deux variables associé à un code de géométrie algébrique présenté ici comme exemple,

- la figure 3 montre les 13 premières lignes de la matrice des syndromes S dans ce même code,

- la figure 4 est un organigramme représentant les étapes initiales d'un procédé de correction d'erreurs selon l'invention,

- les figures 5a et 5b sont des organigrammes représentant les étapes principales suivantes de ce procédé de correction d'erreurs selon l'invention, et

- la figure 6 représente un appareil de réception de signaux numériques incorporant un décodeur selon l'invention.

La **figure 1** est un schéma synoptique d'un système de transmission d'informations utilisant un codage et décodage de canal selon l'invention.

Ce système a pour fonction de transmettre des informations de nature quelconque à partir d'une source 100 vers un destinataire ou utilisateur 109. En premier lieu, la source 100 met ces informations sous la forme de symboles appartenant à un certain alphabet (par exemple des octets de bits), et transmet ces symboles à une unité de stockage 101, qui accumule les symboles de façon à former des ensembles contenant chacun k symboles. Ensuite, chacun de ces ensembles est transmis par l'unité de stockage 101 à un codeur 102 qui y ajoute $(n-k)$ symboles de redondance, de manière à construire un mot du code de longueur n.

Les mots de code ainsi formés sont ensuite transmis à un modulateur 103, qui associe à chaque symbole du mot de code un symbole de modulation (par exemple, une amplitude complexe). Ensuite, ces symboles de modulation sont transmis à un émetteur ou à un enregistreur 104, qui insère les symboles dans un canal de transmission. Ce canal peut être constitué par exemple d'une émission filaire ou non-filaire telle qu'un signal radio, ou par un stockage sur un support adapté tel qu'un DVD ou une bande magnétique. Cette transmission parvient à un récepteur ou à un lecteur 105, après avoir été

affecté par un « bruit de transmission » dont l'effet est de modifier ou d'effacer, aléatoirement, certains des symboles de modulation.

Le récepteur ou lecteur 105 transmet alors ces symboles au démodulateur 106, qui les transforme en symboles de l'alphabet mentionné précédemment, dont chaque ensemble constitue un « mot reçu ». Le mot reçu est ensuite traité par une unité de correction d'erreurs 107, qui met en œuvre un procédé de décodage selon l'invention, de manière à fournir un « mot de code associé ». Puis ce mot de code associé est transmis à une unité de suppression de redondance 108, qui en extrait k symboles d'information en mettant en œuvre un algorithme de décodage inverse de celui mis en œuvre par le codeur 102. Enfin, ces symboles d'information sont fournis à leur destinataire 109.

On peut considérer que les unités 107 et 108 forment conjointement un « décodeur » 10.

On va à présent illustrer le procédé de correction d'erreurs selon l'invention à l'aide d'un exemple numérique. On notera que cet exemple ne constitue pas nécessairement un choix de paramètres préférentiel pour le codage ou le décodage. Il n'est fourni ici que pour permettre à l'homme du métier de comprendre plus facilement le fonctionnement du procédé selon l'invention.

Considérons donc un code de géométrie algébrique de paramètres (512,480) défini comme suit.

L'alphabet des symboles est constitué par les 256 éléments du corps de Galois $\mathbf{F}_{256}$. Chaque élément non-nul de ce corps est égal à une puissance, comprise entre 0 et 254, d'un de ses éléments, noté γ, qui vérifie l'équation

$$\gamma^8 + \gamma^4 + \gamma^3 + \gamma^2 + 1 = 0 ,$$

ce qui implique que : $\gamma^{255} = 1$.

On considère alors la « courbe algébrique » de genre $g = 8$ constituée par l'ensemble des solutions de l'équation à deux inconnues

$$y^2 + y + x^{17} = 0 \qquad (2)$$

sur $\mathbf{F}_{256}$ (cette équation étant de degré 2 en y, elle est dite « hyperelliptique »). Ces solutions, qui sont au nombre de 512, constituent les « points de la courbe », par exemple :

$P_1 = (\gamma^0,\gamma^{85})$, $P_2 = (\gamma^0,\gamma^{170})$, $P_3 = (\gamma^1,\gamma^{119})$, $P_4 = (\gamma^1,\gamma^{153})$, $P_5 = (\gamma^2,\gamma^{51})$,

...

$P_{508} = (\gamma^{253},\gamma^{193})$, $P_{509} = (\gamma^{254},\gamma^{14})$, $P_{510} = (\gamma^{254},\gamma^{224})$, $P_{511} = (0, \gamma^0)$, $P_{512} = (0,0)$.

Chaque point P_i sert à identifier le i-ème élément de tout mot de code ; c'est pourquoi $n = 512$.

Ensuite, on choisit un ensemble de monômes h_i (i =1,...,32) en x et y, dont la liste est donnée en **figure 2**. Ces monômes constituent une base pour un espace vectoriel LF de polynômes en x et y à coefficients dans $\mathbf{F}_{256}$. Le choix des fonctions h_i n'est pas arbitraire, mais obéit à certains critères qui définissent les codes de géométrie algébrique dits « à un point ».

Enfin, la matrice de parité H du code est définie de la manière suivante : l'élément en ligne i et colonne j de cette matrice est égal à la valeur de la fonction h_i au point P_j de la courbe algébrique. Ainsi, $n-k$ = 32, et donc k = 480. Par exemple, compte tenu de $\gamma^{255} = 1$, la douzième ligne de la matrice H est :

$h_{12}(P_1) = \gamma^{85}$, $h_{12}(P_2) = \gamma^{170}$, $h_{12}(P_3) = \gamma^{120}$, $h_{12}(P_4) = \gamma^{154}$,

...

$h_{12}(P_{510}) = \gamma^{223}$, $h_{12}(P_{511}) = 0$, $h_{12}(P_{512}) = 0$,

puisque $h_{12} = xy$.

Le code ayant été choisi, on va montrer à présent comment on construit la matrice des syndromes S.

Considérons les 32 x 32 produits d'une fonction h_i par une fonction h_j, définis modulo l'équation de la courbe algébrique (équation (2)).

Certains de ces produits sont égaux à un élément de l'espace vectoriel LF. Par exemple :

$h_6h_7 = x^5 \cdot x^6 = x^{11} = h_{15}$, $h_{10}h_{10} = y \cdot y = y^2 = x^{17} + y = h_{27} + h_{10}$.

Dans de tels cas, on écrit

$$h_i h_j = \sum_{v=1}^{n-k} \lambda_v h_v \ ,$$

et l'élément S_{ij} de la matrice S est défini par

$$S_{ij} = \sum_{v=1}^{n-k} \lambda_v s_v \ . \qquad (3)$$

L'ordre des fonctions h_i a été choisi de manière à ce que, pour toute valeur de i, le produit de h_i par h_j appartienne à LF pour toutes les valeurs de j comprises entre 1 et une certaine limite $w(i)$, où la fonction $w(i)$ est décroissante. Dans l'exemple numérique considéré (voir la figure 2), $w(i)$ devient inférieur à i à partir de $i = u_{max} = 13$, avec $w(u_{max}) = 12$.

En revanche, certains produits h_ih_j n'appartiennent pas à LF. C'est le cas par exemple de :

$$h_6h_{29} = x^5 \cdot x^{18} = x^{23}, \text{ et de : } h_{17}h_{10} = x^{12} \cdot y .$$

Dans de tels cas, il n'est en fait pas nécessaire de définir la valeur de l'élément S_{ij} correspondant. En effet, l'algorithme selon l'invention, que l'on trouvera décrit ci-dessous, ne fait appel qu'aux éléments S_{ij} pour lesquels j est inférieur ou égal à $w(i)$. Cette fonction w représente donc une « frontière » entre les éléments calculés selon l'équation (3), qui remplissent le coin supérieur gauche de la matrice S, et les éléments indéterminés de cette matrice (voir la **figure 3**, qui représente les 13 premières lignes de la matrice S).

Comme on l'a indiqué plus haut, l'algorithme selon l'invention fait appel à des manipulations linéaires sur les lignes. On rappelle que les manipulations linéaires sur les lignes d'une matrice Σ pour obtenir une matrice Σ' peuvent être représentées par la pré-multiplication de la matrice Σ par une matrice inversible L, soit :

$$\Sigma' = L \cdot \Sigma.$$

Soit μ le nombre de lignes de Σ, et soit L la matrice à μ colonnes qui rend les éléments de la ligne α de Σ' nuls de la colonne $j = 1$ à la colonne $j = w(\mu)$. Autrement dit, L vérifie :

$$0 = \Sigma'_{\alpha j} = (L \cdot \Sigma)_{\alpha j} = \sum_{i=1}^{\mu} L_{\alpha i}\, \Sigma_{ij} \quad \text{pour } j = 1,2,\ldots,w(\mu).$$

Si l'on définit alors

$$l_i \equiv L_{\alpha i} \quad (i = 1,2,\ldots,\mu) , \tag{4}$$

on voit que l'on a obtenu

$$\sum_{i=1}^{\mu} l_i\, \Sigma_{ij} = 0$$

comme souhaité (système d'équations (1)). Ainsi, les nombres l_i cherchés forment la α-ème ligne de L. Or L peut évidemment s'écrire

$$L = L \cdot \mathrm{I} ,$$

où I est la matrice unité, de sorte qu'en pratique on obtient L en appliquant à la matrice unité les mêmes manipulations linéaires de lignes que celles qui sont appliquées à la matrice Σ pour obtenir Σ'.

Les manipulations linéaires mises en œuvre au cours des algorithmes présentés ci-dessous en exemple opèrent, pour les valeurs croissantes de la variable d'itération u, sur la ligne $S[u]$ d'indice u de la matrice S^u, en commençant à $u = 2$. Les mêmes manipulations linéaires sont appliquées successivement à la matrice unité I, dont les lignes résultant de ces manipulations sont notées $L[u]$. La ligne $S[u]$ sur laquelle on opère est remplacée par la somme de cette même ligne et d'une ligne précédente d'indice $i < u$ multipliée par un coefficient approprié ; de manière classique, ces opérations, complétées selon les besoins par des permutations de colonnes, sont aptes à transformer progressivement la matrice S en une matrice triangulaire (avec des zéros dans le coin inférieur gauche). L'algorithme prend fin lorsque l'on trouve une matrice S^λ qui possède une ligne d'indice inférieur ou égal à λ dont les éléments sont nuls sur les $w(\lambda)$ premières colonnes.

On va décrire à présent, en s'appuyant sur les figures 4, 5a et 5b, un premier mode de réalisation de l'invention, dans lequel on calcule λ_0, qui est, rappelons-le, la plus petite valeur de μ pour laquelle le système (1) possède une solution non-triviale. Dans ce mode de réalisation, chaque matrice S^u possède un nombre de colonnes égal à $w(u)$.

La **figure 4** est un organigramme représentant les étapes initiales d'un procédé de correction d'erreurs selon l'invention, qui est mis en œuvre chaque fois que l'unité de correction d'erreurs 107 saisit un nouveau mot reçu (sous forme d'un tableau de symboles appartenant audit alphabet) à l'étape 196 de ce procédé.

On calcule d'abord, à l'étape 197, les syndromes d'erreur du mot reçu, c'est-à-dire les $(n-k)$ composantes s_v du vecteur

$$\underline{s} = H \underline{r}^T ,$$

puis on insère, à l'étape 198, ces syndromes dans la matrice S construite selon l'équation (3) et tronquée à u_{max} lignes (telle que sur la figure 3). La matrice L est initialisée à la matrice unité de dimension u_{max}.

Les entiers u et v représentent respectivement les indices de ligne et de colonne courants. Le rôle de l'entier u^* sera explicité ci-dessous.

On cherche alors à placer le premier pivot de Gauss en position S_{11}. Si la valeur initialement présente est déjà non-nulle, comme vérifié à l'étape 199, on commute (étape 200) vers le sous-algorithme A. Sinon, on commute (étape 300) vers le sous-algorithme B.

Le reste de l'algorithme est constitué d'une série d'étapes, où chaque étape consiste soit à appliquer le sous-algorithme A, soit à appliquer le sous-algorithme B.

Le sous-algorithme A, qui est illustré sur la **figure 5a,** sert essentiellement à descendre d'une ligne dans « S » (étape 201).

L'étape 202 représente un test d'arrêt de l'algorithme qui sera expliqué plus bas. Si l'algorithme doit être poursuivi, on manipule la nouvelle ligne (étape 204) de manière à annuler ses éléments situés sur les $(u-1)$ premières colonnes. On a alors obtenu la matrice S^u pour la valeur u considérée.

A l'étape 205, on compare le nombre courant de lignes u avec le nombre courant de colonnes v. S'il s'avère que u est supérieur à v, on commute vers le sous-algorithme B (étape 300).

Si u est inférieur ou égal à v, on parcourt, à l'étape 206, la ligne u dans les colonnes $j \geq u$, à la recherche du premier élément non-nul.

Si on en trouve un avant la colonne d'indice $(v+1)$, on commence le calcul de la matrice S^{u+1} en échangeant, à l'étape 207, la colonne j sur laquelle se trouve cet élément non-nul avec la colonne u (sauf si $j = u$), afin que cet élément serve de pivot de Gauss en position (u,u), et l'on retourne au point de départ 200 du sous-algorithme A, pour terminer le calcul de S^{u+1} en manipulant la ligne $(u+1)$.

Si en revanche tous les éléments de la ligne u sont nuls jusqu'à la colonne v comprise, on commute (étape 300) vers le sous-algorithme B.

Le sous-algorithme B, qui est représenté sur la **figure 5b**, sert essentiellement à progresser d'une colonne dans « S » (étape 303).

S'il s'avère, au départ de ce sous-algorithme (étape 301), que tous les éléments de la ligne u jusqu'à la colonne $w(u)$ comprise sont nuls, c'est qu'on a alors en fait atteint la fin de l'algorithme (étape 302) : le nombre λ_0 cherché est donc égal à la valeur courante de u.

On notera que, puisqu'au départ $I_{i\lambda_0} = 0$ pour $i < \lambda_0$, les manipulations linéaires des lignes de L d'indice $i \leq \lambda_0$ ne modifient pas les éléments de ces lignes situés sur la colonne λ_0. Par conséquent, dans le cas du critère d'arrêt 301, on aboutit à $I_{\lambda_0} = L_{\lambda_0\lambda_0} = 1$, puisque $I_{\lambda_0\lambda_0} = 1$.

Si en revanche, v est encore inférieur à $w(u)$, on continue à parcourir la ligne u vers les colonnes suivantes (en faisant croître v, étape 303), à la recherche du premier élément non-nul (étape 304).

Si on en trouve un avant, donc, que v ne devienne égal à $w(u)$, on attribue d'abord, à l'étape 305, la valeur courante de u à la variable u^*. Ainsi, la ligne d'indice u^* de S^{u^*} présente un élément non-nul sur sa colonne v, mais tous les éléments de cette ligne situés sur les colonnes 1 à $(v-1)$ sont nuls. La matrice L vérifie donc à ce stade :

$$\sum_{i=1}^{u^*} L_{u^* i}\, S_{ij} = 0 \text{ , pour } j = 1,2,\ldots,(v-1) \text{ ,} \qquad (5)$$

où $L_{u^*u^*}$, notamment, est égal à 1.

Ensuite, à l'étape 306, on commence le calcul de la matrice S^{u+1} en échangeant la colonne v avec la colonne u (sauf si $v = u$), afin que l'élément non-nul trouvé à l'étape 304 serve de pivot de Gauss en position (u,u), et l'on revient au sous-algorithme A, pour terminer le calcul de S^{u+1} en manipulant la ligne $(u+1)$.

Revenons à présent sur le critère d'arrêt 202. Si $v > w(u)$, on en déduit que le nombre λ_0 cherché est égal à la valeur courante de u, et donc : $v-1 \geq w(\lambda_0)$. En effet, si l'on remonte à l'étape 305 qui a précédé cette étape 202, on constate que les éléments de la ligne u^* de S^{u^*} étaient nuls au moins de

la colonne 1 à la colonne $w(\lambda_0)$. Compte tenu de l'équation (5), on voit que si l'on prend :

$$l_i = L_{u^*i} \quad \text{pour } i = 1,2,\ldots,u^* \quad , \text{ et}$$

$$l_i = 0 \quad \text{pour } i = (u^*+1), \ldots,\lambda_0 \ ,$$

on a obtenu effectivement une solution non-triviale du système d'équations (1), où λ_0 correspond à la valeur minimale de u pour laquelle une telle solution existe. On notera que, dans le cas du critère d'arrêt 202, on aboutit à $l_{u^*} = L_{u^*u^*} = 1$. On notera également que dans ce cas il n'a pas été nécessaire de calculer la ligne $S[\lambda_0]$.

Selon un deuxième mode de réalisation de la présente invention, on détermine la « dimension étendue » λ_D visée par l'algorithme modifié étendu. Les étapes de cette variante de l'invention sont essentiellement les mêmes que les étapes du mode de réalisation décrit ci-dessus, si ce n'est que l'on s'abstiendra (le cas échéant) d'incrémenter v au-delà de $w(\mu_D)$; ainsi, le critère d'arrêt 202 devient : $v > \min(w(u), w(\mu_D))$, et le critère d'arrêt 301 devient : $v = \min(w(u), w(\mu_D))$ (où « min » désigne le minimum). Comme mentionné en introduction, cette variante de l'algorithme selon l'invention permet une économie en termes de stockage. Par exemple, pour le code considéré ci-dessus, on trouve $\mu_D = 7$; comme $w(7) = 20$, on aura $w(\lambda_D) \leq 20$, de sorte que l'on n'a besoin que des 20 premières colonnes de la matrice S.

Quel que soit le mode de réalisation de l'invention, on pourra appliquer ses résultats à la correction des erreurs de transmission dans le mot considéré comme si l'on avait mis en œuvre l'algorithme de base modifié ou l'algorithme modifié étendu. Par exemple, selon un procédé de correction d'erreurs connu, on formera le « polynôme de localisation d'erreurs »

$$\Lambda(x,y) \equiv \sum_{i=1}^{\lambda} l_i \, h_i(x,y) \ ,$$

dont les racines servent à trouver la position des erreurs dans le mot reçu (pour plus de détails, on pourra consulter par exemple l'article de Tom Høholdt et Ruud Pellikaan cité ci-dessus). On notera que, pour λ fixé, l'ensemble des coefficients l_i associés n'est pas unique ; par exemple, on peut manifestement

multiplier ici tous les nombres l_i par une même constante sans que cela n'affecte la localisation des erreurs de transmission.

Le schéma synoptique de la **figure 6** représente un appareil de réception de signaux numériques 70 incorporant le décodeur 10. Cet appareil 70 comprend un clavier 711, un écran 709, un destinataire d'informations externe 109, un lecteur de données 105 et un démodulateur 106, conjointement reliés à des ports d'entrée/sortie 703 du décodeur 10 qui est réalisé ici sous la forme d'une unité logique.

Le décodeur 10 comporte, reliés entre eux par un bus d'adresses et de données 702 :

- une unité centrale de traitement 700,
- une mémoire vive (RAM) 704,
- une mémoire morte (ROM) 705, et
- lesdits ports d'entrée/sortie 703.

Chacun des éléments illustrés en figure 7 est bien connu de l'homme du métier des micro-ordinateurs et des systèmes de stockage de masse et, plus généralement, des systèmes de traitement de l'information. Ces éléments connus ne sont donc pas décrits ici. On observe, cependant, que :

- le destinataire d'informations 109 pourrait être, par exemple, un périphérique d'interface, un afficheur, un modulateur, une mémoire externe ou un autre système de traitement de l'information (non représenté), et pourrait être adapté à recevoir des séquences de signaux représentatifs de parole, de messages de service ou de données multimédia notamment de type IP ou ATM, sous forme de séquences de données binaires,
- le lecteur 105 est adapté à lire des données enregistrées sur un support tel qu'un disque magnétique ou magnéto-optique.

La mémoire vive 704 conserve des données, des variables et des résultats intermédiaires de traitement, dans des registres de mémoire portant, dans la description, les mêmes noms que les données dont ils conservent les valeurs. La mémoire vive 704 comporte notamment les registres suivants :

- des registres « *mots_reçus* », dans lesquels sont conservés les mots reçus,

- un registre « *symboles_estimés* », dans lequel sont conservés les symboles issus d'un mot reçu en cours de correction,

- un registre « *mots_associés* », dans lequel sont conservés les symboles des « mots de code associés », et

- un registre « *symboles_information* », dans lequel sont conservés les symboles résultant de la suppression de la redondance.

La mémoire morte 705 est adaptée à conserver, dans des registres qui, par commodité, possèdent les mêmes noms que les données qu'ils conservent :

- le programme de fonctionnement de l'unité centrale de traitement 700, dans un registre « programme »,

- la longueur de chaque mot de code dans un registre « n »,

- le nombre de symboles d'information dans chaque mot de code, dans un registre « k »,

- la liste des valeurs de $w(u)$ pour $1 \leq u \leq u_{max}$, telle que celle illustrée sur la figure 2, dans un registre « w »,

- la liste des coefficients de la matrice des syndromes, telle que celle illustrée sur la figure 3, dans un registre « S », et

- la matrice de parité du code, dans un registre « H ».

On a décrit ci-dessus à titre d'exemple une application de l'invention au stockage de masse des données, mais il est clair que les procédés selon l'invention peuvent tout aussi bien être mis en œuvre au sein d'un réseau de télécommunications, auquel cas l'unité 105 pourrait par exemple être un récepteur adapté à mettre en œuvre un protocole de transmission de données par paquets sur un canal hertzien.

REVENDICATIONS

1. Procédé de décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, dans lequel, afin d'identifier la position des erreurs dans un mot reçu, on définit la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu,

ledit procédé étant caractérisé en ce qu'il comporte des étapes numérotées u de construction de matrices, au cours desquelles l'on construit des matrices S^u en commençant par $S^1 = S$, en ce que chaque matrice S^u pour $u > 1$ est obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis
- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce que la construction de matrices s'arrête lorsque :

- soit $S^u{}_{uj} = 0$ pour tout j compris entre 1 et $w(u)$,
- soit il existe un entier $u^* \leq (u-1)$ tel que $S^{u^*}{}_{u^*j} = 0$ pour tout j compris entre 1 et $w(u)$.

2. Procédé de décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, dans lequel, afin d'identifier la position des erreurs dans un mot reçu, on définit la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu,

ledit procédé étant caractérisé en ce qu'il comporte des étapes numérotées u de construction de matrices, au cours desquelles l'on construit des matrices S^u en commençant par $S^1 = S$, en ce que chaque matrice S^u pour $u > 1$ est obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis

- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce que la dernière étape est :

- soit l'étape de numéro $u = \lambda$, si l'on détermine un entier λ tel que $S^{\lambda}{}_{\lambda j} = 0$ pour tout j compris entre 1 et $w(\lambda)$,

- soit l'étape de numéro $u = (\lambda - 1)$, si l'on détermine un entier λ et un entier u^*, avec $u^* < \lambda$, tels que $S^{u^*}{}_{u^* j} = 0$ pour tout j compris entre 1 et $w(\lambda)$.

3. Procédé de décodage selon la revendication 1 ou la revendication 2, caractérisé en ce que le nombre de lignes de chaque matrice S^u est tronqué à u_{max}, où u_{max} est le plus petit entier i pour lequel $w(i)$ est inférieur à i.

4. Procédé de décodage selon l'une quelconque des revendications 1 à 3, caractérisé en ce que le nombre de colonnes de chaque matrice S^u est tronqué à $w(u)$.

5. Procédé de décodage selon l'une quelconque des revendications 1 à 3, caractérisé en ce que le nombre de colonnes de chaque matrice S^u est tronqué à $w(\mu_D)$ pour u compris entre 1 et le minimum de Duursma μ_D, et à $w(u)$ pour (le cas échéant) u supérieur à μ_D.

6. Dispositif de correction d'erreurs (107) pour le décodage d'un code de géométrie algébrique à un point de dimension k et de longueur n, destiné à identifier la position des erreurs dans un mot reçu, et comprenant des moyens pour définir la matrice des syndromes S, de dimension $(n-k)$ x $(n-k)$, dont les éléments S_{ij} de chaque ligne i sont calculés, pour j compris entre 1 et $w(i)$, où la frontière w est une fonction décroissante, à partir du syndrome $\underline{s}$ de ce mot reçu,

ledit dispositif de correction d'erreurs (107) étant caractérisé en ce qu'il comprend en outre des moyens pour construire des matrices S^u numérotées u, avec $S^1 = S$, chaque matrice S^u pour $u > 1$ étant obtenue à partir de la matrice S^{u-1} en effectuant :

- le cas échéant des permutations sur les colonnes de la matrice S^{u-1}, puis

- des manipulations linéaires sur la ligne d'indice u de la matrice ainsi obtenue,

et en ce qu'il comporte des moyens pour arrêter la construction de matrices lorsque :

- soit $S^u{}_{uj} = 0$ pour tout j compris entre 1 et $w(u)$,
- soit il existe un entier $u^* \leq (u-1)$ tel que $S^{u^*}{}_{u^*j} = 0$ pour tout j compris entre 1 et $w(u)$.

7. Dispositif de correction d'erreurs selon la revendication 6, caractérisé en ce qu'il comprend en outre des moyens pour tronquer à u_{max} le nombre de lignes de chaque matrice S^u, où u_{max} est le plus petit entier i pour lequel $w(i)$ est inférieur à i.

8. Dispositif de correction d'erreurs selon la revendication 6 ou la revendication 7, caractérisé en ce qu'il comprend en outre des moyens pour tronquer à $w(u)$ le nombre de colonnes de chaque matrice S^u.

9. Dispositif de correction d'erreurs selon la revendication 6 ou la revendication 7, caractérisé en ce qu'il comprend en outre des moyens pour tronquer à $w(\mu_D)$ pour u compris entre 1 et le minimum de Duursma μ_D, et à $w(u)$ pour (le cas échéant) u supérieur à μ_D, le nombre de colonnes de chaque matrice S^u.

10. Décodeur (10), caractérisé en ce qu'il comprend :

- au moins un dispositif de correction d'erreurs selon l'une quelconque des revendications 6 à 9, et
- au moins une unité de suppression de la redondance (108).

11. Appareil de réception de signaux numériques codés (70), caractérisé en ce qu'il comprend un décodeur selon la revendication 10, et en ce qu'il comporte des moyens (106) pour démoduler lesdits signaux numériques codés.

12. Système informatique (70), caractérisé en ce qu'il comprend un décodeur selon la revendication 10, et en ce qu'il comprend en outre :

- au moins un disque dur, et
- au moins un moyen de lecture (105) de ce disque dur.

13. Moyen de stockage de données inamovible, caractérisé en ce qu'il comporte des instructions de code de programme informatique pour

l'exécution des étapes d'un procédé selon l'une quelconque des revendications 1 à 5.

14. Moyen de stockage de données partiellement ou totalement amovible, caractérisé en ce qu'il comporte des instructions de code de programme informatique pour l'exécution des étapes d'un procédé selon l'une quelconque des revendications 1 à 5.

15. Programme d'ordinateur, caractérisé en ce qu'il contient des instructions telles que, lorsque ledit programme commande un dispositif de traitement de données programmable, lesdites instructions font que ledit dispositif de traitement de données met en œuvre un procédé selon l'une quelconque des revendications 1 à 5.

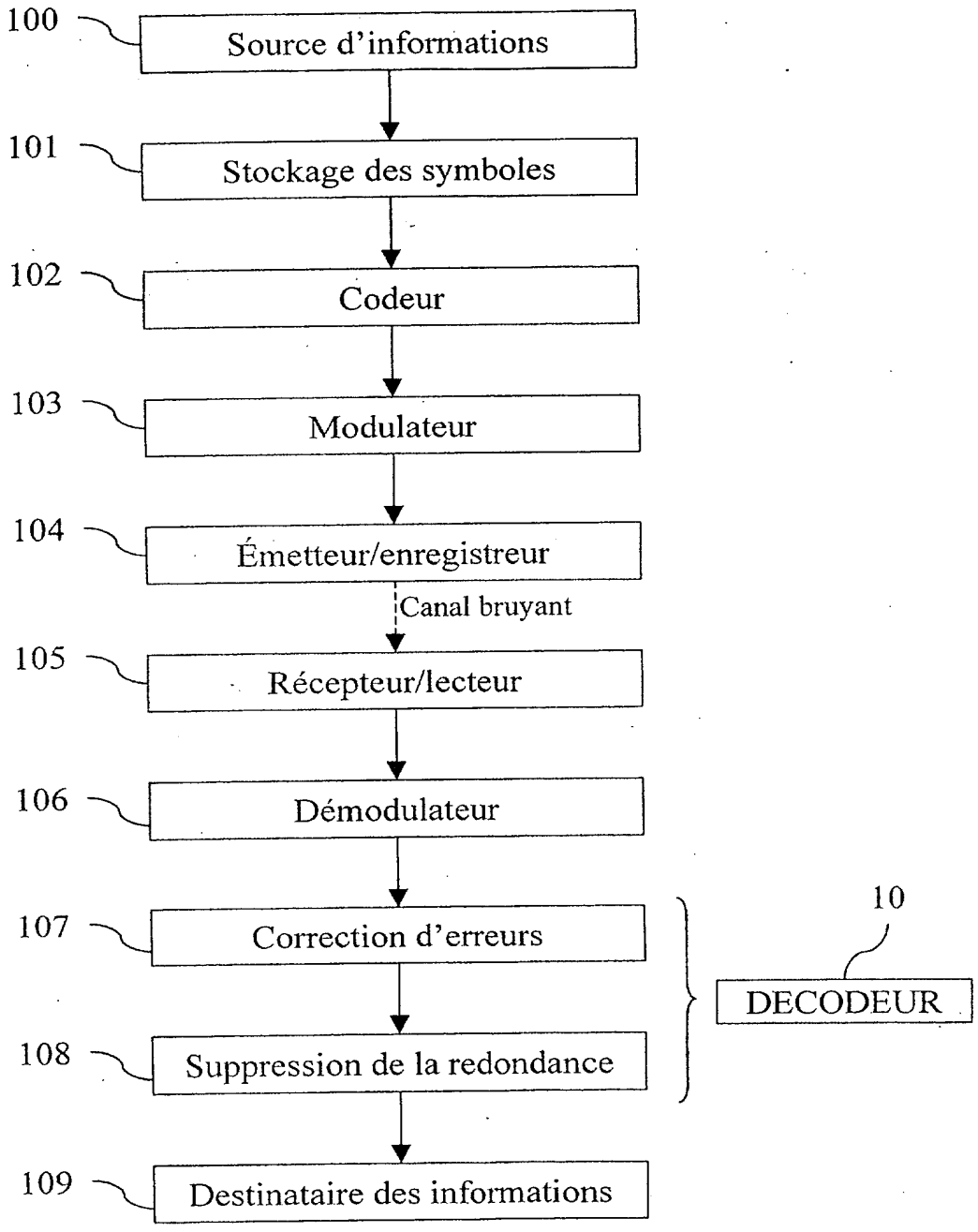
100
Source d'informations
101
Stockage des symboles
102
Codeur
103
Modulateur
104
Émetteur/enregistreur
Canal bruyant
105
Récepteur/lecteur
106
Démodulateur
107
Correction d'erreurs
108
Suppression de la redondance
109
Destinataire des informations
10
DECODEUR

FIG. 1

i	h_i	$w(i)$
1	1	32
2	x	30
3	x^2	28
4	x^3	26
5	x^4	24
6	x^5	22
7	x^6	20
8	x^7	18
9	x^8	16
10	y	15
11	x^9	14
12	xy	13
13	x^{10}	12
14	x^2y	11
15	x^{11}	10
16	x^3y	9
17	x^{12}	8
18	x^4y	8
19	x^{13}	7
20	x^5y	7
21	x^{14}	6
22	x^6y	6
23	x^{15}	5
24	x^7y	5
25	x^{16}	4
26	x^8y	4
27	x^{17}	3
28	x^9y	3
29	x^{18}	2
30	$x^{10}y$	2
31	x^{19}	1
32	$x^{11}y$	1

FIG. 2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}
2	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}		
3	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{11}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}				
4	S_4	S_5	S_6	S_7	S_8	S_9	S_{11}	S_{13}	S_{15}	S_{16}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}						
5	S_5	S_6	S_7	S_8	S_9	S_{11}	S_{13}	S_{15}	S_{17}	S_{18}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}								
6	S_6	S_7	S_8	S_9	S_{11}	S_{13}	S_{15}	S_{17}	S_{19}	S_{20}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}										
7	S_7	S_8	S_9	S_{11}	S_{13}	S_{15}	S_{17}	S_{19}	S_{21}	S_{22}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}												
8	S_8	S_9	S_{11}	S_{13}	S_{15}	S_{17}	S_{19}	S_{21}	S_{23}	S_{24}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}														
9	S_9	S_{11}	S_{13}	S_{15}	S_{17}	S_{19}	S_{21}	S_{23}	S_{25}	S_{26}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}																
10	S_{10}	S_{12}	S_{14}	S_{16}	S_{18}	S_{20}	S_{22}	S_{24}	S_{26}	$S_{27}+S_{10}$	S_{28}	$S_{29}+S_{12}$	S_{30}	$S_{31}+S_{14}$	S_{32}																	
11	S_{11}	S_{13}	S_{15}	S_{17}	S_{19}	S_{21}	S_{23}	S_{25}	S_{27}	S_{28}	S_{29}	S_{30}	S_{31}	S_{32}																		
12	S_{12}	S_{14}	S_{16}	S_{18}	S_{20}	S_{22}	S_{24}	S_{26}	S_{28}	$S_{29}+S_{12}$	S_{30}	$S_{31}+S_{14}$	S_{32}																			
13	S_{13}	S_{15}	S_{17}	S_{19}	S_{21}	S_{23}	S_{25}	S_{27}	S_{29}	S_{30}	S_{31}	S_{32}																				

FIG. 3

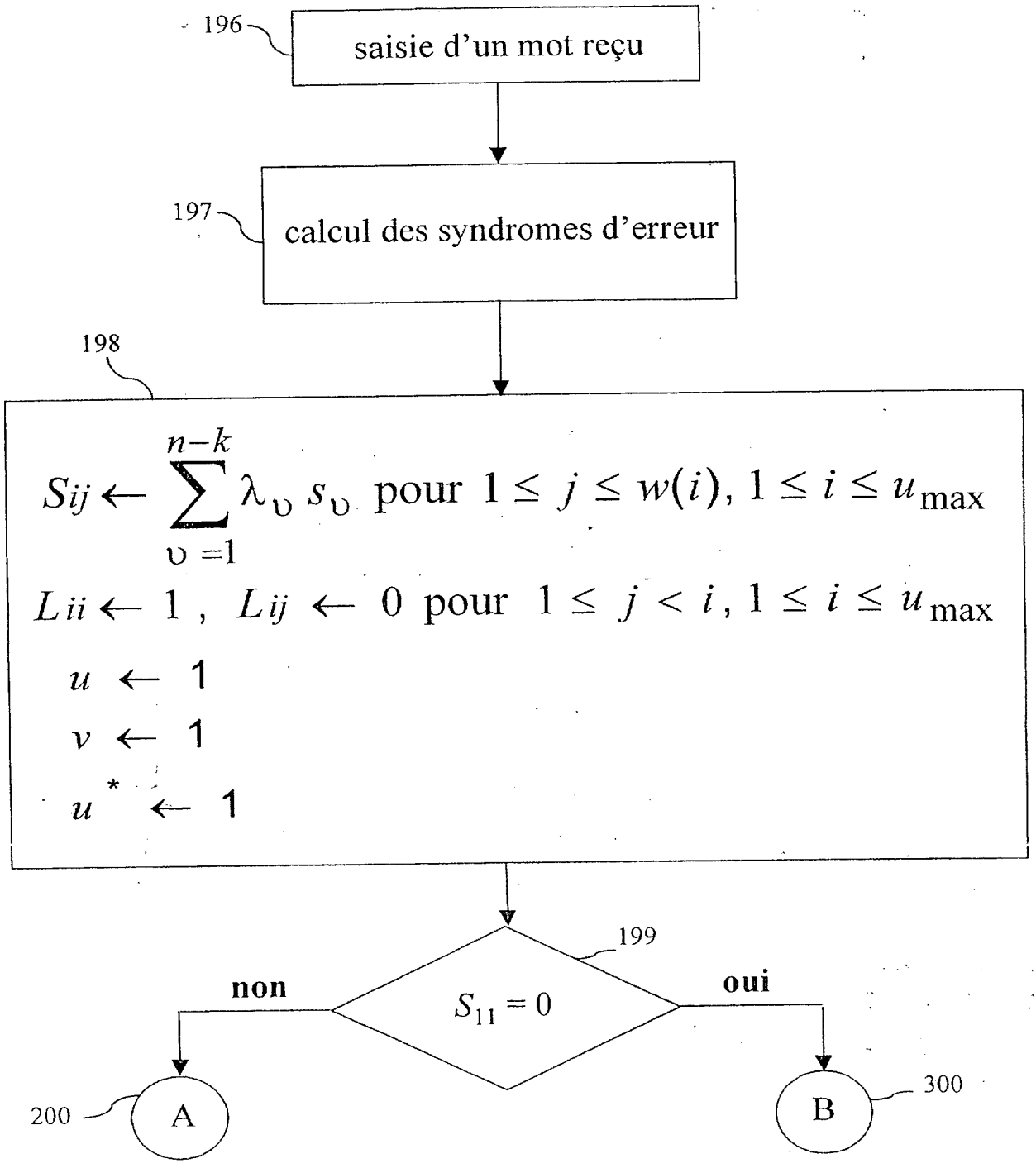
196
saisie d'un mot reçu
197
calcul des syndromes d'erreur
198
$Sij \leftarrow \sum_{\upsilon=1}^{n-k} \lambda_\upsilon s_\upsilon$ pour $1 \leq j \leq w(i), 1 \leq i \leq u_{\max}$
$Lii \leftarrow 1$, $Lij \leftarrow 0$ pour $1 \leq j < i, 1 \leq i \leq u_{\max}$
$u \leftarrow 1$
$v \leftarrow 1$
$u^* \leftarrow 1$
199
$S_{11} = 0$
non
oui
200
A
B
300

FIG. 4

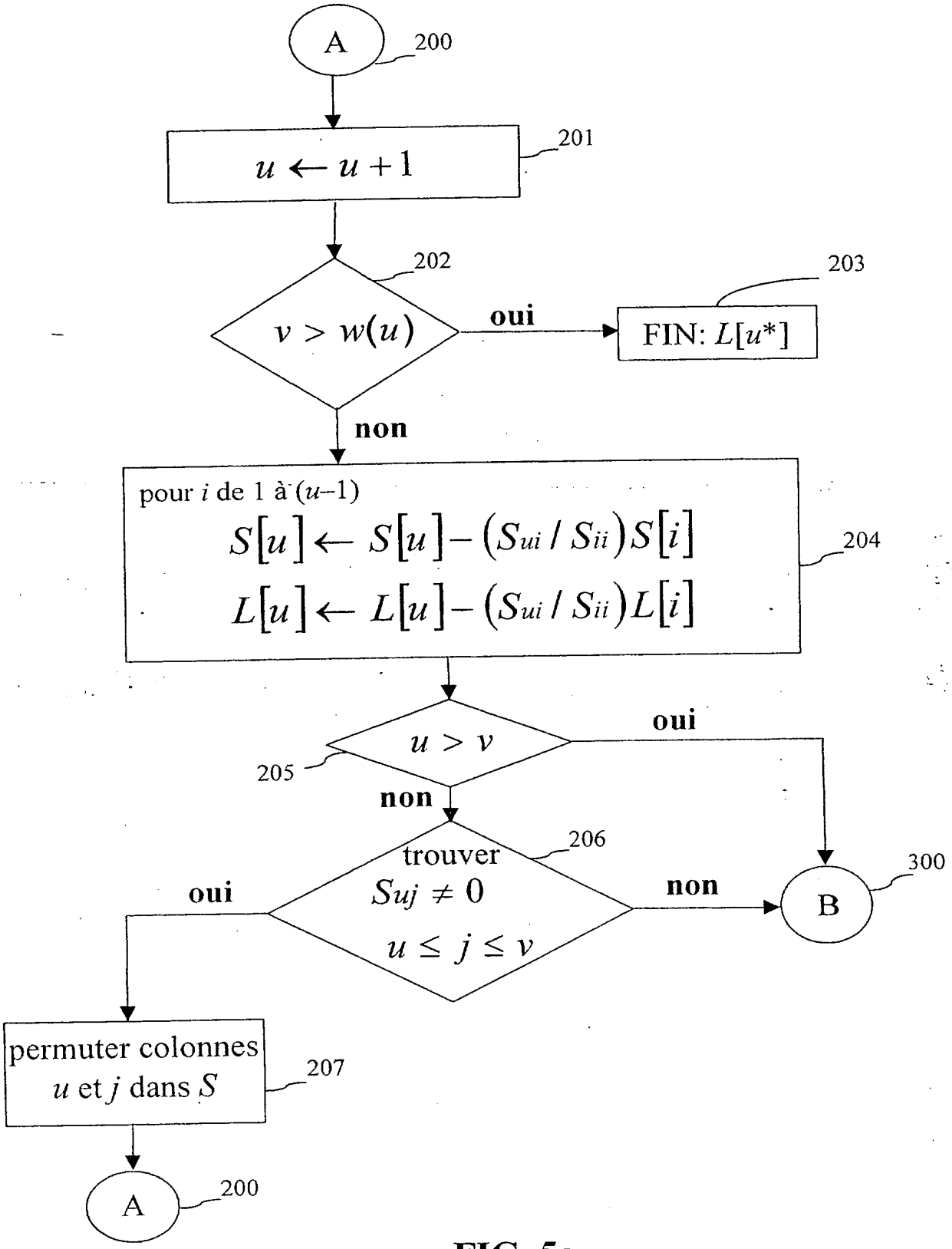
A
200
$u \leftarrow u+1$
201
202
$v > w(u)$
oui
203
FIN: $L[u^*]$
non
pour i de 1 à $(u-1)$
$S[u] \leftarrow S[u] - (S_{ui} / S_{ii}) S[i]$
$L[u] \leftarrow L[u] - (S_{ui} / S_{ii}) L[i]$
204
$u > v$
oui
205
non
trouver
$S_{uj} \neq 0$
$u \leq j \leq v$
206
oui
non
300
B
permuter colonnes
u et j dans S
207
A
200

FIG. 5a

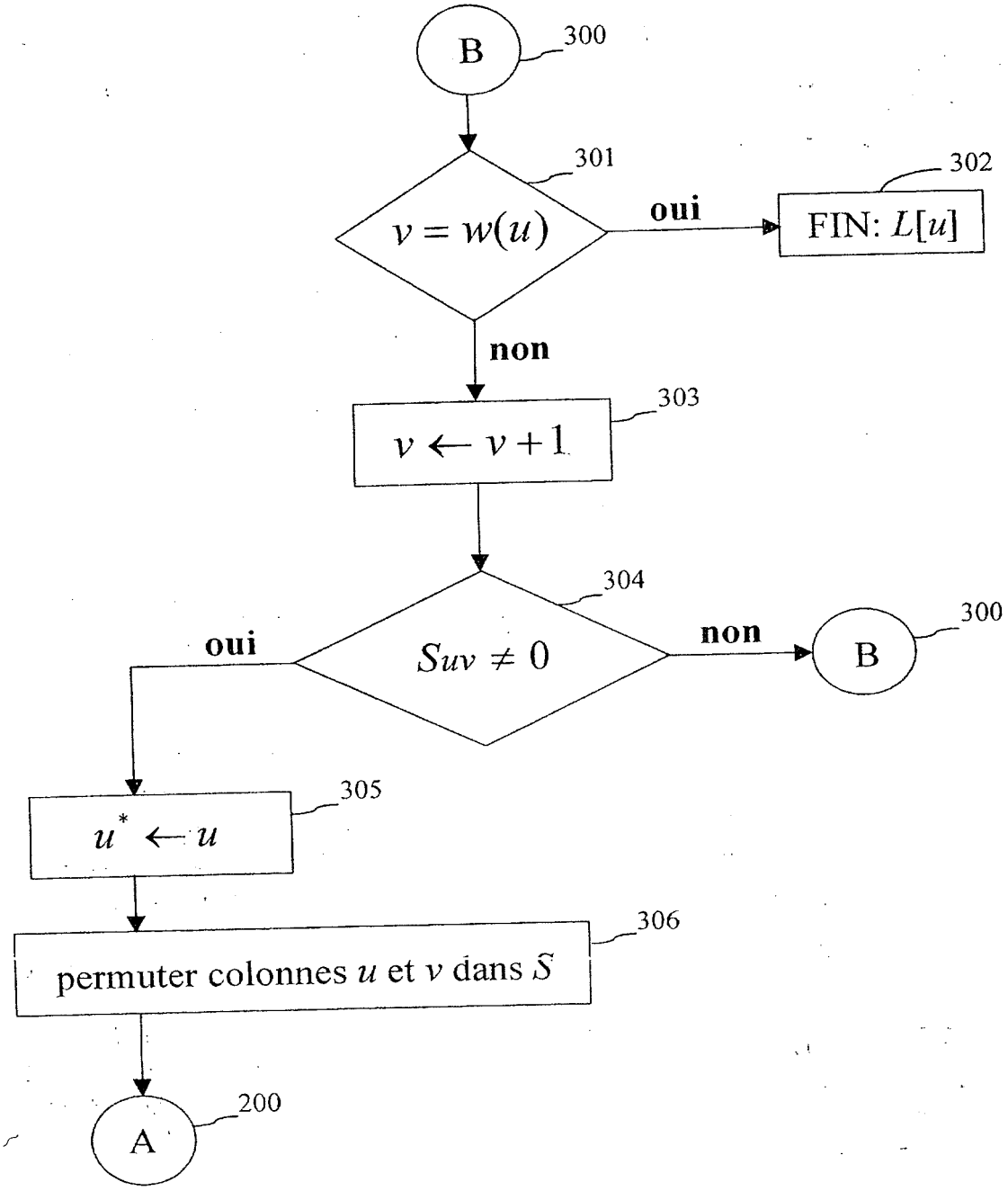
B
300
301
$v = w(u)$
oui
302
FIN: $L[u]$
non
303
$v \leftarrow v+1$
304
$Suv \neq 0$
oui
non
B
300
305
$u^* \leftarrow u$
306
permuter colonnes u et v dans S
A
200

FIG. 5b

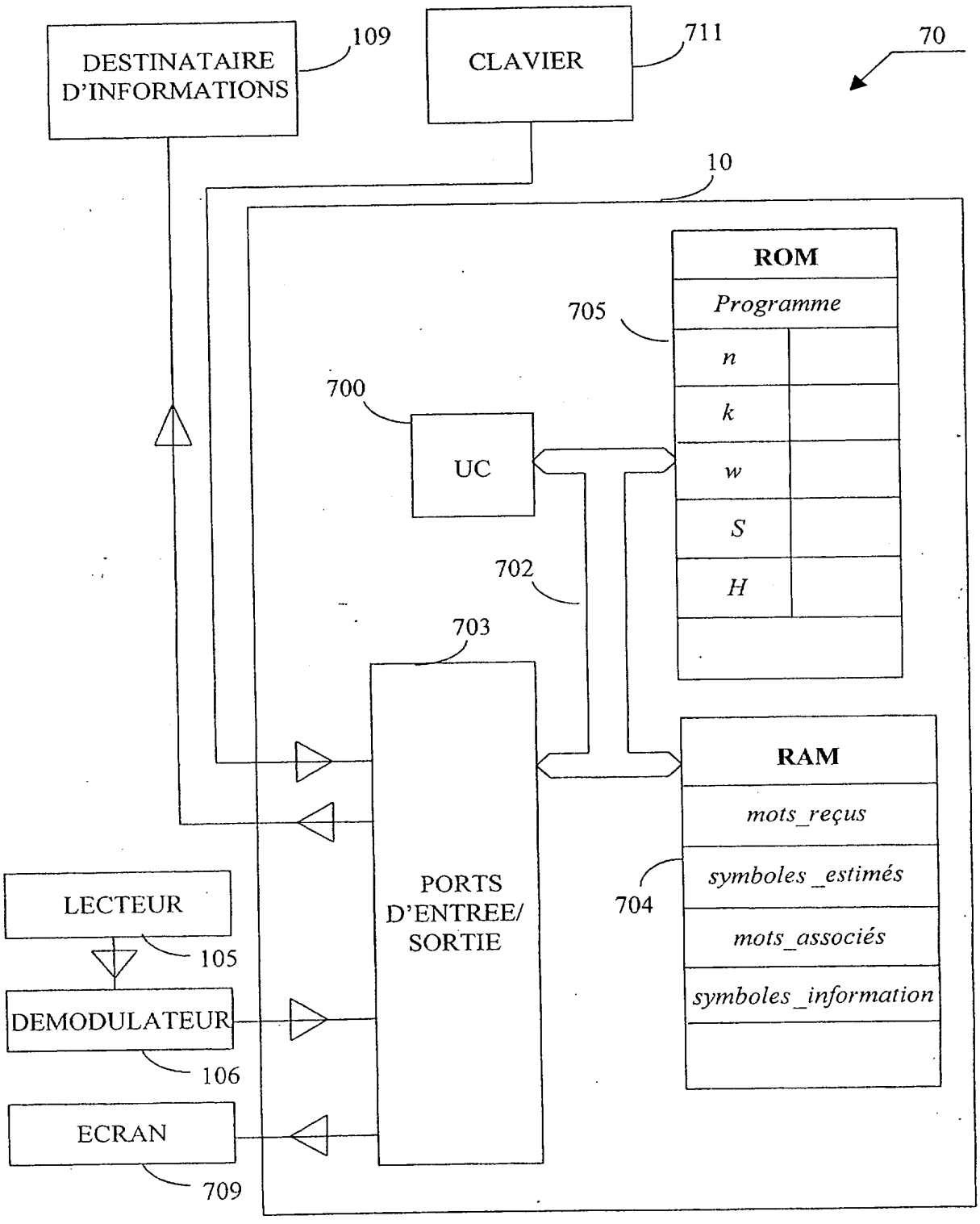
DESTINATAIRE D'INFORMATIONS
109
CLAVIER
711
70
10
ROM
Programme
705
n
k
w
S
H
700
UC
702
703
RAM
mots_reçus
symboles_estimés
704
mots_associés
symboles_information
PORTS D'ENTREE/ SORTIE
LECTEUR
105
DEMODULATEUR
106
ECRAN
709

FIG. 6

reçue le 17/10/02

INPI
INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

DÉPARTEMENT DES BREVETS

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11235*02

DÉSIGNATION D'INVENTEUR(S) Page N° 1/1

(Si le demandeur n'est pas l'inventeur ou l'unique inventeur)

INV

Cet imprimé est à remplir lisiblement à l'encre noire

DB 113 W /300301

Vos références pour ce dossier *(facultatif)*	BIF023147/DM/LJH
N° D'ENREGISTREMENT NATIONAL	0212069

TITRE DE L'INVENTION (200 caractères ou espaces maximum)

Procédés et dispositifs pour le décodage des codes de géométrie algébrique à un point

LE(S) DEMANDEUR(S) :

CANON KABUSHIKI KAISHA

DESIGNE(NT) EN TANT QU'INVENTEUR(S) : (Indiquez en haut à droite «Page N° 1/1» S'il y a plus de trois inventeurs, utilisez un formulaire identique et numérotez chaque page en indiquant le nombre total de pages).

Nom		LEHOBEY
Prénoms		Frédéric
Adresse	Rue	185, rue de Fougères,
	Code postal et ville	35700 RENNES, France
Société d'appartenance *(facultatif)*		
Nom		PIRET
Prénoms		Philippe
Adresse	Rue	4, Boulevard des Métairies,
	Code postal et ville	35510 CESSON-SEVIGNE, France
Société d'appartenance *(facultatif)*		
Nom		
Prénoms		
Adresse	Rue	
	Code postal et ville	
Société d'appartenance *(facultatif)*		

DATE ET SIGNATURE(S) DU (DES) DEMANDEUR(S) OU DU MANDATAIRE (Nom et qualité du signataire)	Le 30 septembre 2002 Bruno QUANTIN N°92.1206 RINUY, SANTARELLI